

Claims

I claim:

Sub A!

1. A computer-implemented method for detecting differences between first and second graphical programs, wherein the method executes on a computer including a display, wherein the first and second graphical programs comprise graphical code, the method comprising:
5 determining differences between said first graphical program and said second graphical program; and
10 displaying an indication of said differences on the display.

2. The method of claim 1, wherein said first and second graphical programs each comprise a block diagram which includes interconnected function icons.

15 3. The method of claim 1, wherein said first and second graphical programs each include a user interface panel which displays data.

20 4. The method of claim 1, wherein said first graphical program includes a first user interface panel for viewing at least one of input to or output from the first graphical program;

wherein said second graphical program includes a second user interface panel for viewing at least one of input to or output from the second graphical program.

25 5. The method of claim 1, wherein said displaying said differences includes displaying a textual description of each of said differences.

30 6. The method of claim 1, wherein said displaying said differences comprises displaying at least one of said first and second graphical programs, wherein said displaying said differences further comprises highlighting at least a subset of said differences between said first and second graphical programs.

7. The method of claim 1, wherein said determining differences between said first graphical program and said second graphical program includes comparing differences between attributes of said first graphical program and attributes of said second graphical program.

8. The method of claim 1, wherein said first and second graphical programs each comprise a block diagram comprising graphical code, a user interface panel for displaying one or more of data output from the block diagram and data input to the block diagram, and graphical program attributes;

wherein said determining differences between said first graphical program and said second graphical program comprises:

determining differences between said block diagrams of said first and second graphical programs;

15 determining differences between said user interface panels of said first and second graphical programs;

determining differences between said graphical program attributes of said first and second graphical programs.

20 9. The method of claim 1, wherein said determining differences between said first and second graphical programs comprises finding functional differences between said first graphical program and said second graphical program, wherein a functional difference indicates a potential difference in execution of said first and second graphical programs.

25

10. The method of claim 9, wherein said determining differences between said first and second graphical programs includes finding cosmetic differences between said first and second graphical programs, wherein a cosmetic difference is a non-functional difference.

30

11. The method of claim 1, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects;

5 wherein said determining differences includes determining differences between said first plurality of objects of said first graphical program and said second plurality of objects of said second graphical program.

10 12. The method of claim 11, wherein said objects include function block icons.

15 13. The method of claim 11, wherein said objects include controls and indicators.

14. The method of claim 11,
wherein said objects include function block icons; and
wherein said objects also include data flow paths coupled between said function block icons.

20 15. The method of claim 11, wherein said determining differences includes matching said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program.

25 16. The method of claim 15, further comprising:
creating data structures to represent said first and second graphical programs,
wherein said matching uses said data structures.

30 17. The method of claim 15, wherein said matching includes determining matches between one or more of said first plurality of objects of said first graphical

program with one or more of said second plurality of objects of said second graphical program.

18. The method of claim 17, wherein said matching comprises determining 5 objects of said first graphical program which have a same type as objects of said second graphical program.

19. The method of claim 18, wherein said matching further comprises determining if an object of said first graphical program which has the same type as an 10 object of said second graphical program has the same type as another object in said second graphical program.

20. The method of claim 15, wherein said determining differences includes determining objects present in said first graphical program which have no matching 15 object in said second graphical program.

21. The method of claim 15, wherein said determining differences includes comparing matching objects in said first graphical program and said second graphical program to determine differences in said matching objects.

20
21. The method of claim 21, wherein each of said objects comprises a set of attributes and methods, wherein said comparing matching objects includes comparing attributes of said matching objects in said first graphical program and said second graphical program.

25
23. The method of claim 15,
wherein, for each respective object in said first graphical program, said matching comprises:

30 comparing the respective object in said first graphical program with one or more objects in said second graphical program;

5 determining a score value for a plurality of object pairs in response to said comparing, wherein each of said plurality of object pairs comprises two object elements, said two object elements comprising said respective object in said first graphical program and an object of said one or more objects in said second graphical program, wherein said score value indicates a degree of matching between said object elements comprising said object pair.

10 24. The method of claim 23, wherein said determining said score value for one of said plurality of object pairs includes examining connectivity of object elements comprising said object pair.

15 25. The method of claim 23, wherein said determining said score value for one of said plurality of object pairs includes examining attributes of object elements comprising said object pair.

20 26. The method of claim 23, wherein said score value is a weighted score value, wherein said determining said score value for one of said plurality of object pairs includes examining connectivity of object elements comprising said object pair and examining attributes of object elements comprising said object pair;

25 wherein said connectivity of object elements has greater weight than said attributes of said object elements.

27. The method of claim 26, wherein said attributes comprise functional attributes and cosmetic attributes;

25 wherein said functional attributes have greater weight than said cosmetic attributes.

30 28. The method of claim 23, wherein said matching further comprises: storing said score values in a match matrix, wherein said match matrix comprises rows which correspond to said first plurality of objects in said first graphical program,

and wherein said match matrix comprises columns which correspond to said second plurality of objects in said second graphical program;

wherein for an ith row and a jth column, the matrix element (i,j) includes the score value for the object pair comprising object i and object j.

5

29. The method of claim 28, wherein said matching further comprises:
resolving said match matrix using said match matrix score values, wherein said resolving produces a 1:1 or 1:0 relationship between objects in said first graphical program and objects in said second graphical program depending upon whether or not an
10 object in said first graphical program has a matching object in said second graphical program.

30. The method of claim 28, wherein said matching further comprises:
resolving said match matrix using said match matrix score values, wherein, for
15 each of at least a subset of said rows corresponding to said first plurality of objects, said resolving produces a column from said second plurality of objects in said second graphical program, wherein said produced column has a match matrix element with a highest score value for the respective row.

20 31. The method of claim 15,

wherein said matching produces a resolved match matrix, wherein said resolved match matrix comprises matching information regarding matches between said first plurality of objects in said first graphical program and said second plurality of objects in said second graphical program;

25 wherein said determining differences between said first graphical program and said second graphical program comprises:

grouping objects which have exact matches with connectivity matches into exact matching connected components using said matching information;

30 grouping remaining objects to produce non-exact matching connected components wherein said grouping remaining objects produces a first list of non-exact-

matching connected components and produces a second list of non-matching connected components;

5 matching said first and second lists of non-exact matching connected components, wherein said matching said first and second lists of non-exact matching connected components includes finding common objects in said first and second lists of non-exact matching connected components, wherein said matching said first and second lists of non-exact matching connected components produces second matching information;

10 merging said first and second lists of non-exact matching connected components using said second matching information, wherein said merging produces a composite non-exact matching connected component list;

storing corresponding non-exact matching connected components in a result data structure;

15 comparing matching objects in corresponding non-exact matching connected components to determine additional object differences;

storing said additional object differences in said result data structure.

32. The method of claim 31, wherein said grouping objects which have exact matches with connectivity matches comprises grouping objects which have exact matches 20 with connectivity matches using said matching information to produce a first list of exact matching connected components and to produce a second list of exact matching connected components, wherein said grouping objects includes comparing objects according to one or more criteria to determine exact matches between objects, wherein an exact match comprises a match in the match matrix and match in said comparing.

25

33. The method of claim 15,

wherein said first graphical program comprises a first user interface panel including a subset of said first plurality of objects;

30 wherein said second graphical program comprises a second user interface panel including a subset of said second plurality of objects;

wherein said matching produces a user interface panel match matrix comprising matching information;

wherein said determining differences includes determining differences between said first user interface panel and said second user interface panel which comprises:

5 comparing matching top-level objects of said first and second user interface panels to determine differences between said top-level objects using said matching information;

comparing low-level objects of said top-level objects to determine differences between said low-level objects using said matching information;

10 storing said differences in response to said comparing matching top-level objects of said first and second user interface panels and said comparing low-level objects of said top-level objects.

34. The method of claim 11, further comprising:

15 creating one or more data structures which includes information regarding said first plurality of objects in said first graphical program and said second plurality of objects in said second graphical program;

wherein said determining differences uses said one or more data structures.

20 35. The method of claim 34, wherein said one or more data structures each comprise directed graphs comprising a plurality of vertices connected by edges, wherein each of said vertices represents one of said objects in a respective one of said graphical programs.

25 Sub a³ 36. A computer-implemented method for detecting differences between first and second graphical programs, wherein the method executes on a computer including a display, the method comprising:

30 creating the first graphical program, wherein the first graphical program comprises first graphical code;

creating the second graphical program, wherein the second graphical program comprises second graphical code;
determining differences between said first graphical program and said second graphical program; and
5 displaying an indication of said differences on the display.

~~43~~

~~42~~

37. The method of claim 36,

wherein the first graphical program comprises a first plurality of objects;
wherein the second graphical program comprises a second plurality of objects;
10 wherein said determining differences comprises determining differences between said first plurality of objects in said first graphical program and said second plurality of objects in said second graphical program.

~~44~~

~~42~~

15 38. The method of claim 36, wherein said graphical code includes function

block icons.

~~45~~

~~42~~

39. The method of claim 36, wherein said graphical code includes interconnected function block icons.

~~46~~

~~42~~

20 40. The method of claim 36, wherein said creating the second graphical program comprises modifying the first graphical program to create the second graphical program.

Sub 04

25 41. A computer-implemented method for detecting differences between first and second graphical programs, wherein the method executes on a computer including a display, the method comprising:

30 creating the first graphical program, wherein the first graphical program comprises first graphical code, wherein the first graphical program comprises a first plurality of objects;

creating the second graphical program, wherein the second graphical program comprises second graphical code, wherein the second graphical program comprises a second plurality of objects;

5 determining differences between said first plurality of objects in said first graphical program and said second plurality of objects in said second graphical program; and

displaying an indication of said differences on the display.

42. A computer-implemented method for detecting differences between first 10 and second graphical programs, wherein the method executes on a computer including a display, the method comprising:

creating the first graphical program, wherein the first graphical program comprises first graphical code, wherein the first graphical program comprises a first plurality of objects;

15 creating the second graphical program, wherein the second graphical program comprises second graphical code, wherein the second graphical program comprises a second plurality of objects;

matching said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program;

20 determining differences between said first graphical program and said second graphical program in response to said matching; and

displaying an indication of said differences on the display.

43. A computer-implemented method for detecting differences between first 25 and second graphical programs, wherein the method executes on a computer including a display, wherein the first and second graphical programs comprise graphical code, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects, the method comprising:

matching said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program;

determining differences between said first graphical program and said second graphical program in response to said matching; and

displaying an indication of said differences on the display screen.

44. A computer system for detecting differences between first and second graphical programs, comprising:

a central processing unit (CPU);

a display coupled to the CPU; and

a memory medium coupled to the CPU which stores said first and second graphical programs, wherein each of the said first and second graphical programs comprises graphical code, wherein the memory medium also stores a difference program for detecting differences between said first and second graphical programs;

wherein said CPU is operable to execute said difference program to determine differences between said first graphical program and said second graphical program and to display an indication of said differences on the display.

20 ~~51~~ 45. The computer system of claim ~~44~~, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects;

wherein said CPU is operable to match said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program to determine differences between said first graphical program and said second graphical program.

25 ~~52~~ 46. The computer system of claim ~~45~~, wherein said objects include function block icons.

~~53~~ 47 The computer system of claim ~~45~~, wherein said objects include controls and indicators for displaying data.

5

~~54~~ 48 The computer system of claim ~~45~~,
wherein said objects include function block icons; and
wherein said objects also include data flow paths coupled between said function block icons.

10

~~55~~ 49 The computer system of claim ~~44~~, wherein, to display an indication of said differences on the display, the CPU is operable to display at least one of said first and second graphical programs and highlight at least a subset of said differences between said first and second graphical programs.

20

~~56 Q5~~ 50. A computer system for detecting differences between first and second graphical programs, comprising:

25

a central processing unit (CPU);

a display coupled to the CPU; and

30

a memory medium coupled to the CPU which stores said first and second graphical programs, wherein each of the said first and second graphical programs comprises graphical code, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects, wherein the memory medium also stores a difference program for detecting differences between said first and second graphical programs;

25

wherein said CPU is operable to execute said difference program to:

match said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program;

determine differences between said first graphical program and said second graphical program in response to the matching; and

30

display an indication of said differences on the display.

51. A computer-readable storage medium comprising program instructions for detecting differences between first and second graphical programs, wherein the first and second graphical programs each comprise graphical code, wherein the program instructions execute on a computer including a display, wherein the program instructions are executable to implement:

5 10 determining differences between said first graphical program and said second graphical program; and

10 displaying an indication of said differences on the display.

52. ⁵⁸ The memory medium of claim ⁵¹, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects;

15 wherein said determining differences includes determining differences between said first plurality of objects of said first graphical program and said second plurality of objects of said second graphical program.

53. ⁵⁹ The memory medium of claim ⁵², wherein said objects include function 20 block icons.

54. ⁵⁸ The memory medium of claim ⁵², wherein said objects include controls and indicators.

55. ⁶¹ The memory medium of claim ⁵²,
25 wherein said objects include function block icons; and
wherein said objects also include data flow paths coupled between said function block icons.

62
56. The memory medium of claim 51, wherein said first and second graphical programs each comprise a block diagram which includes interconnected function icons.

63
57
5 56. The memory medium of claim 51, wherein said first and second graphical programs each include a user interface panel which displays data.

64
58
57
58. The memory medium of claim 51, wherein said displaying said differences includes displaying a textual description of each of said differences.

65
59
57
10 59. The memory medium of claim 51, wherein said displaying said differences comprises displaying at least one of said first and second graphical programs, wherein said displaying said differences further comprises highlighting at least a subset of said differences between said first and second graphical programs.

66
60
57
15 60. The memory medium of claim 51, wherein said determining differences between said first graphical program and said second graphical program includes comparing differences between attributes of said first graphical program and attributes of said second graphical program.

67
61
57
20 61. The memory medium of claim 51, wherein said first and second graphical programs each comprise a block diagram comprising graphical code, a user interface panel for displaying one or more of data output from the block diagram and data input to the block diagram, and graphical program attributes;

25 wherein said determining differences between said first graphical program and said second graphical program comprises:

determining differences between said block diagrams of said first and second graphical programs;

determining differences between said user interface panels of said first and second graphical programs;

determining differences between said graphical program attributes of said first and second graphical programs.

68 57
5 62. The memory medium of claim 51, wherein said determining differences between said first and second graphical programs comprises finding functional differences between said first graphical program and said second graphical program, wherein a functional difference indicates a potential difference in execution of said first and second graphical programs.

69 68
10 63. The memory medium of claim 62, wherein said determining differences between said first and second graphical programs includes finding cosmetic differences between said first and second graphical programs, wherein a cosmetic difference is a non-functional difference.

70 57
15 64. The memory medium of claim 51, wherein the first graphical program comprises a first plurality of objects and wherein the second graphical program comprises a second plurality of objects;

wherein said determining differences includes matching said first plurality of objects of said first graphical program with said second plurality of objects of said second graphical program.

71 70
25 65. The memory medium of claim 64, wherein said matching includes determining matches between one or more of said first plurality of objects of said first graphical program with one or more of said second plurality of objects of said second graphical program.

72 71
66. The memory medium of claim 65, wherein said matching comprises determining objects of said first graphical program which have a same type as objects of said second graphical program.

73 67. The memory medium of claim 66, wherein said matching further comprises determining if an object of said first graphical program which has the same type as an object of said second graphical program has the same type as another object in said second graphical program.

5 74 68. The memory medium of claim 64, wherein said determining differences includes determining objects present in said first graphical program which have no matching object in said second graphical program.

10 75 69. The memory medium of claim 64, wherein said determining differences includes comparing matching objects in said first graphical program and said second graphical program to determine differences in said matching objects.

15 76 70. The memory medium of claim 69, wherein each of said objects comprises a set of attributes and methods, wherein said comparing matching objects includes comparing attributes of said matching objects in said first graphical program and said second graphical program.

20 77 71. The memory medium of claim 64, wherein said first and second graphical programs each comprise a block diagram comprising graphical code, a user interface panel for displaying one or more of data output from the block diagram and data input to the block diagram, and graphical program attributes;

wherein said determining differences between said first graphical program and said second graphical program comprises:

25 determining differences between said block diagrams of said first and second graphical programs;

determining differences between said user interface panels of said first and second graphical programs;

30 determining differences between said graphical program attributes of said first and second graphical programs.

78
72 The memory medium of claim 64, wherein, for each respective object in said first graphical program, said matching comprises:

5 comparing the respective object in said first graphical program with one or more objects in said second graphical program;

10 determining a score value for a plurality of object pairs in response to said comparing, wherein each of said plurality of object pairs comprises two object elements, said two object elements comprising said respective object in said first graphical program and an object of said one or more objects in said second graphical program, wherein said score value indicates a degree of matching between said object elements comprising said object pair.

79
73 The memory medium of claim 72, wherein said determining said score value for one of said plurality of object pairs includes examining connectivity of object elements comprising said object pair.

80
74 The memory medium of claim 72, wherein said determining said score value for one of said plurality of object pairs includes examining attributes of object elements comprising said object pair.

20 81
75 The memory medium of claim 72, wherein said score value is a weighted score value, wherein said determining said score value for one of said plurality of object pairs includes examining connectivity of object elements comprising said object pair and examining attributes of object elements comprising said object pair;

25 wherein said connectivity of object elements has greater weight than said attributes of said object elements.

82
76 The memory medium of claim 75, wherein said attributes comprise functional attributes and cosmetic attributes;

wherein said functional attributes have greater weight than said cosmetic attributes.

~~83~~ 78 The memory medium of claim ~~72~~, wherein said matching further 5 comprises:

storing said score values in a match matrix, wherein said match matrix comprises rows which correspond to said first plurality of objects in said first graphical program, and wherein said match matrix comprises columns which correspond to said second plurality of objects in said second graphical program;

10 wherein for an ith row and a jth column, the matrix element (i,j) includes the score value for the object pair comprising object i and object j.

~~84~~ 78 The memory medium of claim ~~72~~, wherein said matching further comprises:

15 resolving said match matrix using said match matrix score values, wherein said resolving produces a 1:1 or 1:0 relationship between objects in said first graphical program and objects in said second graphical program depending upon whether or not an object in said first graphical program has a matching object in said second graphical program.

20 ~~85~~ 78 The memory medium of claim ~~72~~, wherein said matching further comprises:

25 resolving said match matrix using said match matrix score values, wherein, for each of at least a subset of said rows corresponding to said first plurality of objects, said resolving produces a column from said second plurality of objects in said second graphical program, wherein said produced column has a match matrix element with a highest score value for the respective row.

~~86~~ 70 The memory medium of claim ~~64~~,

wherein said first graphical program comprises a first user interface panel including a subset of said first plurality of objects for displaying one or more of data output and data input;

5 wherein said second graphical program comprises a second user interface panel including a subset of said second plurality of objects for displaying one or more of data output and data input;

 wherein said matching produces a user interface panel match matrix comprising matching information;

 wherein said determining differences includes determining differences between 10 said first user interface panel and said second user interface panel which comprises:

 comparing matching top-level objects of said first and second user interface panels to determine differences between said top-level objects using said matching information;

15 comparing low-level objects of said top-level objects to determine differences between said low-level objects using said matching information;

 storing said differences in response to said comparing matching top-level objects of said first and second user interface panels and said comparing low-level objects of said top-level objects.

²⁰
add a⁶ >